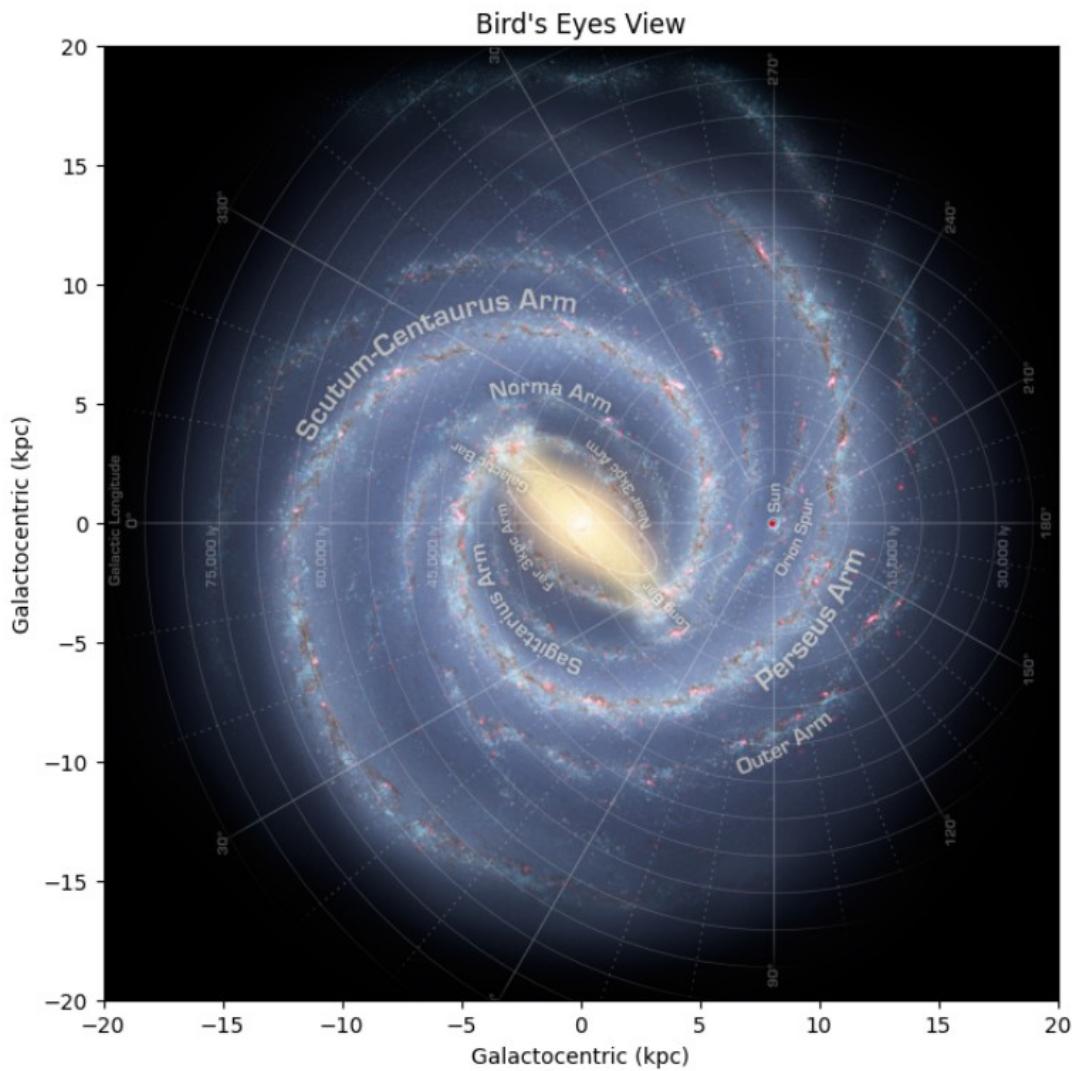


## **Disclaimer**

*I strongly invite you not to use LLMS (chatGPT or so) for this task. Please do not make me read code generated, or even just polished by a bot, it would be a missed opportunity for you to learn how to approach a problem and how to code it (that means, to be really able to solve it). I look forward to seeing how much you learn in the class, what has been clear and what has not been clear.*

## **Assignment 4 (Deadline 5. December 23:59)**

### **TOPIC 1: Unsupervised learning: Clustering the Milky Way**



*Task 0: Install the python library from the page: [milkyway-plot.readthedocs.io](http://milkyway-plot.readthedocs.io) [5]*

*(from a jupyter notebook, run in a cell !pip install mw\_plot Use a python environment! Ask the Tas in case of doubts)*

*Task 1: Install the git package and reproduce the image of the milky way via MW SkyMap (see attached code). [5]*

*Task 2: Generate a few (at least three in total) visualizations of the milky way sector starting in different **centers** (try "M31") and with **different radius** (be careful on the units!). [5]*

*Task 3: Convert the image generated into a `rgb np.array` (each pixel will be a list of 3 number, Red, Green, Blue (`rbg`)). [5]*

*Task 4: Ideate, describe in words and generate a set of categories from the task 3 data (e.g. red? Grey?). This is your **encoding**. [5]*

*Task 5: Use K-NN (Nearest neighbor, not discussed in class but extremely simple) or K-means to **cluster** the data [5]*

*Task 6: Over-impose your cluster to the image generated in task 2 [5]*

*Task 7: Try different categories (task 4), repeating tasks 5 and 6, and provide an explanation of your results. What can we learn from an unsupervised model of our galaxy? [10]*

#### **TIPS:**

*Here you will really benefit to construct a set of functions that can be re-used over and over again. Once done, you can promptly change the **centers** and the **radius** of the observation window, select the **encoding**, perform **clustering**. In the discussion part, remember the limit of unsupervised approaches.*

*I made a quick-and-dirty jupyter notebook for you to exemplify the more technical data handling part. I just generated one encoding, you have thus an example of pipeline to connect to the notebook I shared in class. The idea is for you to focus on the method application and results interpretation.*

**TOPIC 2: Supervised learning: Machines versus human models, who can save the world??**



*Task 0: Reproduce the project 2 exercise 5 plots for Ebola epidemic in West Africa (data points and model prediction). If you are not confident from previous results, ask the TAs (or another group) for the correct solution [5]*

*Task 1: Train a line with linear regression on the data for the three countries. [5]*

*Task 2: Train a better fitting function than a single line with linear regression on the data for the three countries. [10]*

*Task 3: Train a NN network and predict the epidemic evolution. Careful here in your training/test split, remember the assumptions you need to take here. [10]*

*Task 4: Train a LSTM (a NN specialized for time series) and predict the epidemic evolution. [15] (More points here available because I did not provide a code for this, this is a bit of self-learning task, I suggest to refer to <https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/>).*

*Task 5: Discuss the results. Can we ignore modeling and let Machine Learning make prediction? What is a good prediction for these cases? What are your conclusions? [15]*

**TIPS:**

*Here the work is to adopt a set of ML tools. Make sure that sklearn and tensorflow are properly installed. Using an environment with python 3.13 will make this easier. If you are having problem in the installation of these library, make a new environment and re-install them.*

**Deliverable:**

*Please put all the functions (and classes) you have generated in a .py file.*

*In a jupyter notebook, instead, import and use your functions (and classes) to calculate the desired results.*

*In the notebook, provide a short introduction in which you outline the nature of the problem(s) to be investigated.*

*End your notebook with a brief summary of what you feel you learned from the project (if anything). Also, if you have any general comments or suggestions for what could be improved in future assignments, this is the place to do it.*

*Your notebook has to be working.*

*Any figures you include should be easily understandable. You should label axes appropriately, and depending on the problem, include other legends etc. Also, you should discuss your figures in the main text.*

*It is always good if you can reflect a little bit around why you see what you see.*

**Delivery method:**

*Git: you should already have your repo. Make a new folder for project 4 and push I (move your previous assignments in dedicated folders).*

*In a new jupyter notebook, use **TOPIC 1**: task 1, task 2, etc to give the answer to each point using Markdown syntax. Please use, for each task, one cell to help us going over your work effectively.*

**How will this project be graded:**

*45 + 60 points available. Each question has a number of points available indicated in [].*

*Your final score = total points accumulated + quality\_factor \* bonus points*

*quality\_factor = total points accumulated/max point available*

*If AI is used the quality factor will be equal to zero. If AI is used and not stated, it is considered cheating and the whole project will receive 0 points.*

A grade for the assignment between A and F will be given.

**Bonus points:**

*Use **pylint** on your function file. Pylint gives a score from 0 to 10. That will be your bonus points that will be applied.*

*Up to 5 extra points for proper use of **assert** statements in your code. Just writing assert True (or something very trivial like that) won't grant you points.*

*1 extra point for each function that tests if your code works correctly (max 5 points).*

*The shorter the jupyter notebook in terms of number of lines, the more bonus points you will get (Up to 5 points). Comments and assert statements do not count in the line numbers.*

*For each day earlier that you deliver, you will get **one** extra points. For each day of delay, -10. Be strategic about this. (I reduced the relative importance in this task as some of you have other exams ongoing).*

**Group work:**

*Groups of 3 are suggested. If you do not have a group, you will be assigned one. If you are unsatisfied with your group, we can help you to find another one, but we need to be informed within 48hrs from the start of the present project.*

**Other Requirements**

*A statement on the eventual usage of AI has to be included.*

*A statement reporting the Individual contributions has to be written.*

*A self reflection statement at the end it is a precious feedback for us.*